```
ISIS-II PL/M-86 V2.0 COMPILATION OF MODULE PSEUDOM
OBJECT MODULE PLACED IN PSEUD2.OBJ
COMPILER INVOKED BY:  :F0:PSEUD2.PLM DEBUG OPTIMIZE(2) DATE(FEBRUARY 2, 1982) XREF


            $title ('PSEUDO INSTRUCTION MODULE-2')
  1         pseudom:
            do;

            /*

              modified  3/28/81  R. Silberstein
              modified  4/1/81   R. Silberstein
              modified  4/9/81   R. Silberstein
              modified  4/15/81  R. Silberstein
              modified  7/24/81  R. Silberstein
              modified  9/2/81   R. Silberstein

            */

            /*
                    This is the module to perform the decoding of
                    all legal pseudo instructions of the assembler.
                    There is one subroutine for each corresponding
                    pseudoinstruction.
            */

            $include (:f1:macro.lit)
          =  $nolist
            $include (:f1:struc.lit)
          =  $nolist
            $include (:f1:equals.lit)
          =  $nolist
            $include (:f1:ermod.lit)
          =  $nolist
            $include (:f1:files.ext)
          =  $nolist
            $include (:f1:subr1.ext)
          =  $nolist
            $include (:f1:subr2.ext)
          =  $nolist
            $include (:f1:scan.ext)
          =  $nolist
            $include (:f1:print.ext)
          =  $nolist
            $include (:f1:expr.ext)
          =  $nolist
            $include (:f1:ermod.ext)
          =  $nolist
            $include (:f1:pseud2.x86)
          =  $nolist
```

CP/M-86 v:1.1 (vol. IV)
COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. # C81-162

```
            $eject
            /*************** COMMON SUBROUTINES *************/

            /* routine to test if rest of line is either a comment or empty -
              if not, print error message - skip rest of line */

 139   1    test$emptyline: proc;
 140   2      if not emptyline then call errmsg(end$of$line$err);
 142   2      call skip$rest$of$line;
 143   2    end test$emptyline;

            /* perform handling for PAGEWIDTH- and PAGESIZE-routine */

 144   1    sizewidth: proc(p);
 145   2      dcl oper operandstruc at (.operands(0)),p address,dest based p byte;
 146   2      if pass=0 then$do
 148   3        call skip$rest$of$line;   /* do nothing in pass 0 */
 149   3      else$do
 151   3        if expression(.oper) then$do
 153   4          if oper.stype = number then$do
 155   5            dest=oper.offset;
 156   5            call test$emptyline;
 157   5            return;
 158   5          end$if;
                 end$if;
 160   3        call errmsg(pseudooperr);
 161   3        call skip$rest$of$line;
 162   3      end$if;
 163   2    end sizewidth;
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
                $eject

                /****************** PSEUDO SUBROUTINES **************/

164   1         IFrout: proc public;
165   2           DECLARE IFNESTMAX LIT '5';   /* MAX LEVEL OF IF NEXTING */
166   2           dcl oper operandstruc at (.operands(0)),bool byte;

167   2           IFerr: proc;
168   3             call errmsg(ifparerr);
169   3             call skip$rest$of$line;
170   3           end IFerr;

171   2           skip$until$ENDIF: proc;
172   3             dcl pseudotype byte at (.token.value);
173   3             DECLARE LOCIFLEVEL BYTE;
174   3             DECLARE TEMP BYTE;
175   3             LOCIFLEVEL = IFLEVEL + 1;
176   3             IF PRINTON AND NOT IFLIST THEN$DO
178   4               CALL PRINTSOURCELINE;
179   4               PRINTON = FALSE;
180   4               TEMP = TRUE;
181   4             ELSE$DO
183   4               TEMP = FALSE;
184   4             END$IF;
185   3             do while not eofset;  /* (forever) */
186   4               call scan;
187   4               IF TOKEN.TYPE = PSEUDO THEN$DO
189   5                 IF PSEUDOTYPE = PENDIF THEN$DO
191   6                   LOCIFLEVEL = LOCIFLEVEL - 1;
192   6                   IF LOCIFLEVEL = 0 THEN$DO
194   7                     CALL SCAN;
195   7                     CALL TESTEMPTYLINE;
196   7                     IF TEMP THEN PRINTON = TRUE;
198   7                     RETURN;
199   7                   END$IF;
                        ELSE$DO
202   6                   IF PSEUDOTYPE = PIF THEN$DO
204   7                     LOCIFLEVEL = LOCIFLEVEL + 1;
205   7                   END$IF;
                        END$IF;
207   5                 END$IF;
208   4                 call skip$rest$of$line;
209   4               end$while;
210   3           end skip$until$ENDIF;

211   2           IF IFLEVEL = IFNESTMAX THEN$DO
213   3             call errmsg(nestediferr);
214   3             call skip$rest$of$line;
215   3           else$do
217   3             if not noforwardexpr(.oper) then$do
219   4               call IFerr;
220   4             else$do
222   4               if oper.stype <> number then$do
224   5                 call IFerr;
225   5               else$do
227   5                 bool=oper.offset;
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
220  5            if bool <> 0 then$do
230  6              IFLEVEL = IFLEVEL + 1;
231  6              call test$emptyline;
232  6            else$do
234  6              call skip$rest$of$line;
235  6              call skip$until$ENDIF;
236  6            end$if;
237  5          end$if;
238  4        end$if;
239  3      end$if;
240  2    end IFrout;

241  1    ENDIFrout: proc public;
242  2      IF IFLEVEL > 0 THEN$DO
244  3        IFLEVEL = IFLEVEL - 1;
245  3        call test$emptyline;
246  3      else$do
248  3        call errmsg(missiferr);
249  3        call skip$rest$of$line;
250  3      end$if;
251  2    end ENDIFrout;

252  1    INCLUDErout: proc public;
253  2      dcl (disk,i,errflag) byte,filname(11) byte,filtype(3) byte at (.filname (8));

254  2      syntaxerr: proc;
255  3        call errmsg(filesynterr);
256  3        errflag=true;
257  3      end syntaxerr;

258  2      accum$not$alpha: proc byte;
259  3        i=0ffh;
260  3        do while (i:=i+1) < acclen;
261  4          if not alphanumeric(accum(i)) then return true;
           end$while;
264  3        return false;
265  3      end accum$not$alpha;

266  2      if include$on then$do
268  3        call errmsg(nestedincludeerr);
269  3        call skip$rest$of$line;
270  3        return;
271  3      end$if;

272  2      errflag=false;
273  2      disk=include$default;  /* default disk is current one */
274  2      CALL FILL (SPACE, SIZE (FILNAME), .FILNAME);

275  2      if (acclen=1) and (nextch=':') and (letter(accum(0))) then$do

          /* disk name found */
277  3        disk=accum(0)-'A';
278  3        call scan;          /* skip : */
279  3        call scan;          /* get filename */
280  3      end$if;

          /* test syntax of filename */
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
281  2       if (acclen > 8) or accum$not$alpha then$do
283  3         call syntaxerr;      /* illegal filename */
284  3       else$do
286  3         call copy(acclen,.accum(0),.filname); /* pick up filename */
287  3         call scan;                            /* skip filename */

               /* test if filetype - if so, pick it up */
288  3         if specialtoken(',') then$do
290  4           call scan;          /* skip , */
291  4           if (acclen > 3) or accum$not$alpha then$do
293  5             call syntaxerr;
294  5           else$do
296  5             call copy(acclen,.accum(0),.filtype(0));
297  5             call scan;
298  5           end$if;
299  4         ELSE$DO
301  4           CALL COPY (3, .('A86'), .FILTYPE);   /* DEFAULT FILE TYPE */
302  4         end$if;
303  3       end$if;

304  2       if errflag then$do
306  3         call skip$rest$of$line;
307  3       else$do
               /* try to open include file */
309  3         call i$file$setup(disk,.filname,.filtype);
310  3         CALL OPEN$INCLUDE;
311  3         call test$emptyline;
312  3         include$on=true;
313  3       end$if;
314  2     end INCLUDErout;

315  1     ENDrout: proc public;
316  2       call test$emptyline;
317  2       eofset=true;
318  2     end ENDrout;

319  1     PAGESIZErout: proc public;
320  2       call sizewidth(.pagesize);
321  2     end PAGESIZErout;

322  1     PAGEWIDTHrout: proc public;
323  2       call sizewidth(.maxcol);
324  2     end PAGEWIDTHrout;

325  1     TITLErout: proc public;
326  2       do case pass;
327  3         do;          /* pass 0 */
328  4         if token.type=string then$do
330  5           call fill(0,length(title),.title(0));
331  5           if acclen > length(title) then acclen=length(title);
333  5           call copy(acclen,.accum(0),.title(0));
334  5         end$if;
335  4         call skip$rest$of$line;
336  4         end;
337  3         do;          /* do nothing in pass 1 */
338  4         call skip$rest$of$line;
339  4         end;
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
 340  3          do;          /* pass 2 */
 341  4          if token.type=string then$do
 343  5            call scan;
 344  5            call test$emptyline;
 345  5          else$do
 347  5            call errmsg(pseudooperr);
 348  5            call skip$rest$of$line;
 349  5          end$if;
 350  4          end;
 351  3        end$case;
 352  2      end TITLErout;

 353  1      EJECTrout: proc public;
 354  2        if print$on then call printnewpage;
 356  2        call test$emptyline;
 357  2      end EJECTrout;

 358  1      SIMFORMrout: proc public;
 359  2        simform=true;
 360  2        call test$emptyline;
 361  2      end SIMFORMrout;

 362  1      LISTrout: proc public;
 363  2        call test$emptyline;
 364  2        if printswitchoff then$do
 366  3          printswitchoff=false;
 367  3          print$on=true;
 368  3          sourceptr=0;
 369  3        end$if;
 370  2      end LISTrout;

 371  1      NOLISTrout: proc public;
 372  2        if print$on then$do
 374  3          call test$emptyline;
 375  3          call printsourceline;
 376  3          printswitchoff=true;
 377  3          print$on=false;
 378  3        else$do
 380  3          call test$emptyline;
 381  3        end$if;
 382  2      end NOLISTrout;

 383  1      IFLISTROUT: PROC PUBLIC;
 384  2        CALL TESTEMPTYLINE;
 385  2        IFLIST = TRUE;
 386  2      END IFLISTROUT;

 387  1      NOIFLISTROUT: PROC PUBLIC;
 388  2        CALL TESTEMPTYLINE;
 389  2        IFLIST = FALSE;
 390  2      END NOIFLISTROUT;

 391  1      end$module pseudom;
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

CROSS-REFERENCE LISTING

DEFN	ADDR	SIZE	NAME, ATTRIBUTES, AND REFERENCES
138	0000H	1	ACCLEN . . . . . BYTE EXTERNAL(62) 260 275 281 286 291 296 331 332 333
138	0000H	80	ACCUM. . . . . BYTE ARRAY(80) EXTERNAL(63) 261 275 277 286 296 333
258	0240H	47	ACCUMNOTALPHA. . . PROCEDURE BYTE STACK=0006H 281 291
2			ADDR . . . . . LITERALLY 46 51 54 57 60 66 84 87 90 93 138 145 166
74	0000H		ALPHANUMERIC . . . PROCEDURE BYTE EXTERNAL(26) STACK=0000H 261
77	0000H		ASCIICHAR. . . . . PROCEDURE BYTE EXTERNAL(27) STACK=0000H
166	0003H	1	BASEINDEX. . . . . BYTE MEMBER(OPER)
138	0008H	1	BASEINDEX. . . . . BYTE MEMBER(OPERANDS)
145	0003H	1	BASEINDEX. . . . . BYTE MEMBER(OPER)
3			BASEREGBIT . . . . LITERALLY
3			BASEREGCOUNT . . . LITERALLY
166	0000H	1	BOOL . . . . . BYTE 227 228
3			BREGBIT. . . . . LITERALLY
3			BREGCOUNT. . . . . LITERALLY
5			BYT. . . . . LITERALLY
16	0000H	1	CH . . . . . BYTE PARAMETER 17
65	0000H	1	CH . . . . . BYTE PARAMETER 66
13	0000H	1	CH . . . . . BYTE PARAMETER 14
10	0000H	1	CH . . . . . BYTE PARAMETER 11
71	0000H	1	CH . . . . . BYTE PARAMETER 72
68	0000H	1	CH . . . . . BYTE PARAMETER 69
109	0000H	1	CH . . . . . BYTE PARAMETER 110
74	0000H	1	CH . . . . . BYTE PARAMETER 75
77	0000H	1	CH . . . . . BYTE PARAMETER 78
80	0000H	1	CH . . . . . BYTE PARAMETER 81
39	0000H		CLOSEHEX . . . . . PROCEDURE EXTERNAL(13) STACK=0000H
37	0000H		CLOSEINCLUDE . . . PROCEDURE EXTERNAL(12) STACK=0000H
41	0000H		CLOSEPRINT . . . . PROCEDURE EXTERNAL(14) STACK=0000H
33	0000H		CLOSESOURCE. . . . PROCEDURE EXTERNAL(10) STACK=0000H
43	0000H		CLOSESYMBOL. . . . PROCEDURE EXTERNAL(15) STACK=0000H
4			CODE . . . . . LITERALLY
9			CODEMACROERR . . . LITERALLY
3			CODEMACROHEAD. . . LITERALLY
56	0000H		COPY . . . . . PROCEDURE EXTERNAL(20) STACK=0000H 286 296 301 333
2			CR . . . . . LITERALLY
83	0000H	2	D. . . . . WORD PARAMETER 84
86	0000H	2	D. . . . . WORD PARAMETER 87
56	0000H	2	D. . . . . WORD PARAMETER 57
89	0000H	2	D. . . . . WORD PARAMETER 90
92	0000H	2	D. . . . . WORD PARAMETER 93
2			DCL. . . . . LITERALLY
92	0000H		DECOUT . . . . . PROCEDURE EXTERNAL(32) STACK=0000H
4			DELETEDSYMB. . . . LITERALLY
138	0001H	1	DESCR. . . . . BYTE MEMBER(TOKEN)
145	0000H	1	DEST . . . . . BYTE BASED(P) 155
45	0000H	1	DEV. . . . . BYTE PARAMETER 46
68	0000H		DIGIT. . . . . PROCEDURE BYTE EXTERNAL(24) STACK=0000H
253	0003H	1	DISK . . . . . BYTE 273 277 309

COPYRIGHT - 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. =

```
  4             DOUBLEDEFINED. . .   LITERALLY
  9             DOUBLEDEFLAB . . .   LITERALLY
  9             DOUBLEDEFSYMB. . .   LITERALLY
  9             DOUBLEDEFVAR . . .   LITERALLY
  5             DWRD . . . . . .     LITERALLY
123  0000H      EJECT. . . . . .     PROCEDURE EXTERNAL(45) STACK=0000H
353  02F5H  18  EJECTROUT. . . . .   PROCEDURE PUBLIC STACK=000AH
  2             ELSEDO . . . . . .   LITERALLY       199
107  0000H      EMPTYLINE. . . . .   PROCEDURE BYTE EXTERNAL(38) STACK=0000H       140
  2             ENDCASE. . . . . .   LITERALLY
  2             ENDDO. . . . . .     LITERALLY
  2             ENDFOREVER . . . .   LITERALLY
  2             ENDIF. . . . . .     LITERALLY       158  205
241  0119H  30  ENDIFROUT. . . . .   PROCEDURE PUBLIC STACK=000AH
  2             ENDMODULE. . . . .   LITERALLY
  2             ENDOFFILE. . . . .   LITERALLY
  9             ENDOFLINEERR . . .   LITERALLY       141
  2             ENDPROC. . . . . .   LITERALLY
315  026FH  13  ENDROUT. . . . . .   PROCEDURE PUBLIC STACK=000AH
  2             ENDWHILE . . . . .   LITERALLY       262
138  0000H   1  EOFSET . . . . . .   BYTE EXTERNAL(64)        185  317
 83  0000H      EQUAL. . . . . .     PROCEDURE BYTE EXTERNAL(29) STACK=0000H
253  0005H   1  ERRFLAG. . . . . .   BYTE       256  272  304
135  0000H      ERRMSG . . . . . .   PROCEDURE EXTERNAL(50) STACK=0000H       141  160  168  213  248  255
                                      268  347
135  0000H   1  ERRNO. . . . . .     BYTE PARAMETER       136
  4             ERROR. . . . . .     LITERALLY
129  0000H      EXPRESSION . . . .   PROCEDURE BYTE EXTERNAL(48) STACK=0000H       151
  2             FALSE. . . . . .     LITERALLY       179  183  264  272  366  377  389
 62  0000H   2  FCBADR . . . . . .   WORD PARAMETER       63
 62  0000H      FILEABORT. . . . .   PROCEDURE EXTERNAL(22) STACK=0000H
  3             FILEISTRUCTURE . .   LITERALLY
  3             FILEOSTRUCTURE . .   LITERALLY
 48  0000H      FILESETUP. . . . .   PROCEDURE BYTE EXTERNAL(17) STACK=0000H
  9             FILESYNTERR. . . .   LITERALLY       255
 65  0000H      FILL . . . . . .     PROCEDURE EXTERNAL(23) STACK=0000H       274  330
 45  0000H   2  FILNAM . . . . . .   WORD PARAMETER       46
253  0006H  11  FILNAME. . . . . .   BYTE ARRAY(11)       253  274  286  309
 45  0000H   2  FILTYP . . . . . .   WORD PARAMETER       46
253  000EH   3  FILTYPE. . . . . .   BYTE ARRAY(3) AT         296  301  309
  9             FIRSTITEM. . . . .   LITERALLY
  2             FOREVER. . . . . .   LITERALLY
  2             FORMFEED . . . . .   LITERALLY
 86  0000H      HEX1OUT. . . . . .   PROCEDURE EXTERNAL(30) STACK=0000H
 89  0000H      HEX2OUT. . . . . .   PROCEDURE EXTERNAL(31) STACK=0000H
253  0004H   1  I. . . . . . . .     BYTE       259  260  261
  4             IDENT. . . . . .     LITERALLY
167  0097H  14  IFERR. . . . . .     PROCEDURE STACK=0006H         219  224
 45  0000H      IFILESETUP . . . .   PROCEDURE EXTERNAL(16) STACK=0000H       309
133  0000H   1  IFLEVEL. . . . . .   BYTE EXTERNAL(66)        175  211  230  242  244
138  0000H   1  IFLIST . . . . . .   BYTE EXTERNAL(54)        176  385  389
383  0353H  13  IFLISTROUT . . . .   PROCEDURE PUBLIC STACK=000AH
165             IFNESTMAX. . . . .   LITERALLY       211
  9             IFPARERR . . . . .   LITERALLY       168
164  004DH  76  IFROUT . . . . . .   PROCEDURE PUBLIC STACK=000EH
  9             ILLEGALMACRO . . .   LITERALLY
  9             ILLEGALPSEUDO. . .   LITERALLY
```

```
  9             ILLEXPRELEM. . . .   LITERALLY
  9             ILLIOPER . . . . .   LITERALLY
138 0000H   1   INCLUDEDEFAULT . .   BYTE EXTERNAL(68)        273
138 0000H   1   INCLUDEON. . . . .   BYTE EXTERNAL(65)        266  312
252 0137H 249   INCLUDEROUT. . . .   PROCEDURE PUBLIC STACK=000CH
  3             INDEXREGBIT. . . .   LITERALLY
  3             INDEXREGCOUNT. . .   LITERALLY
 21 0000H       ININCLUDEBYTE. . .   PROCEDURE BYTE EXTERNAL(4) STACK=0000H
  2             INIT . . . . . .     LITERALLY
 19 0000H       INSOURCEBYTE . . .   PROCEDURE BYTE EXTERNAL(3) STACK=0000H
  9             INSTRERR . . . . .   LITERALLY
  3             IREGBIT. . . . . .   LITERALLY
  3             IREGCOUNT. . . . .   LITERALLY
  4             LAB. . . . . .       LITERALLY
  9             LABOUTOFRANGE. . .   LITERALLY
  8             LEFTBRACKET. . . .   LITERALLY
145 0000H   2   LENGTH . . . . . .   WORD MEMBER(OPER)
138 0000H   2   LENGTH . . . . . .   WORD MEMBER(OPERANDS)
166 0000H   2   LENGTH . . . . . .   WORD MEMBER(OPER)
              LENGTH . . . . . .   BUILTIN        330  331  332   .
 71 0000H       LETTER . . . . . .   PROCEDURE BYTE EXTERNAL(25) STACK=0000H      275
  2             LF . . . . . . . .   LITERALLY
362 0314H  30   LISTROUT . . . . .   PROCEDURE PUBLIC STACK=000AH
  2             LIT. . . . . . . .   LITERALLY        2    3    4    5    6    7    8    9  165
173 0001H   1   LOCIFLEVEL . . . .   BYTE       175  191  192  204
138 0000H   1   MAXCOL . . . . . .   BYTE EXTERNAL(55)        323
  9             MISSIFERR. . . . .   LITERALLY       248
  9             MISSINGPSEUDO. . .   LITERALLY
  9             MISSINSTR. . . . .   LITERALLY
  9             MISSSEGMINFO . . .   LITERALLY
  9             MISSTYPEINFO . . .   LITERALLY
 92 0000H   2   N. . . . . .         WORD PARAMETER        93
 89 0000H   2   N. . . . . .         WORD PARAMETER        90
 86 0000H   1   N. . . . . .         BYTE PARAMETER        87
 83 0000H   1   N. . . . . .         BYTE PARAMETER        84
 65 0000H   1   N. . . . . .         BYTE PARAMETER        66
 56 0000H   1   N. . . . . .         BYTE PARAMETER        57
 53 0000H   2   N. . . . . .         WORD PARAMETER        54
  4             NEGLECTED. . . . .   LITERALLY
  9             NEGLECTERR . . . .   LITERALLY
  9             NESTEDIFERR. . . .   LITERALLY       213
  9             NESTEDINCLUDEERR .   LITERALLY       268
138 0000H   1   NEXTCH . . . . . .   BYTE EXTERNAL(61)        275
  5             NIL. . . . . .       LITERALLY
132 0000H       NOFORWARDEXPR. . .   PROCEDURE BYTE EXTERNAL(49) STACK=0000H      217
127 0000H       NOFORWARDOPER. . .   PROCEDURE BYTE EXTERNAL(47) STACK=0000H
387 0360H  13   NOIFLISTROUT . . .   PROCEDURE PUBLIC STACK=000AH
371 0332H  33   NOLISTROUT . . . .   PROCEDURE PUBLIC STACK=000AH
  3             NOOVERCOUNT. . . .   LITERALLY
  3             NOOVERRIDEBIT. . .   LITERALLY
  4             NUMBER . . . . . .   LITERALLY       153  222
  8             OAND . . . . . .     LITERALLY
  8             OEQ. . . . . .       LITERALLY
145 0006H   2   OFFSET . . . . . .   WORD MEMBER(OPER)        155
138 0006H   2   OFFSET . . . . . .   WORD MEMBER(OPERANDS)
166 0006H   2   OFFSET . . . . . .   WORD MEMBER(OPER)        227
  8             OGE. . . . . .       LITERALLY
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
  8              OGT. . . . . .   LITERALLY
  8              OLAST. . . . . .   LITERALLY
  8              OLE. . . . . .   LITERALLY
  8              OLENGTH. . . . .   LITERALLY
  8              OLT. . . . . .   LITERALLY
  8              OMOD . . . . . .   LITERALLY
  8              ONE. . . . . .   LITERALLY
  8              ONOT . . . . . .   LITERALLY
  8              OOFFSET. . . . .   LITERALLY
  8              OOR. . . . . .   LITERALLY
 27  0000H       OPENHEX. . . . .   PROCEDURE EXTERNAL(7) STACK=0000H
 25  0000H       OPENINCLUDE. . .   PROCEDURE EXTERNAL(6) STACK=0000H        310
 29  0000H       OPENPRINT. . . .   PROCEDURE EXTERNAL(8) STACK=0000H
 23  0000H       OPENSOURCE . . .   PROCEDURE EXTERNAL(5) STACK=0000H
 31  0000H       OPENSYMBOL . . .   PROCEDURE EXTERNAL(9) STACK=0000H
145  0000H   9   OPER . . . . . .   STRUCTURE EXTERNAL(67) AT      151  153  155
166  0000H   9   OPER . . . . . .   STRUCTURE EXTERNAL(67) AT      217  222  227
125  0000H       OPERAND. . . . .   PROCEDURE BYTE EXTERNAL(46) STACK=0000H
138  0000H  36   OPERANDS . . . .   STRUCTURE ARRAY(4) EXTERNAL(67)          145  166
  3              OPERANDSTRUC . .   LITERALLY       138  145  166
  4              OPERATOR . . . .   LITERALLY
  9              OPMISMATCH . . .   LITERALLY
  8              OPTR . . . . . .   LITERALLY
  8              OSEG . . . . . .   LITERALLY
  8              OSHL . . . . . .   LITERALLY
  8              OSHORT . . . . .   LITERALLY
  8              OSHR . . . . . .   LITERALLY
  8              OTYPE. . . . . .   LITERALLY
 10  0000H       OUTHEXBYTE . . .   PROCEDURE EXTERNAL(0) STACK=0000H
 13  0000H       OUTPRINTBYTE . .   PROCEDURE EXTERNAL(1) STACK=0000H
 16  0000H       OUTSYMBOLBYTE. .   PROCEDURE EXTERNAL(2) STACK=0000H
 59  0000H       OUTTEXT. . . . .   PROCEDURE EXTERNAL(21) STACK=0000H
  8              OXOR . . . . . .   LITERALLY
144  0004H   2   P. . . . . .   WORD PARAMETER AUTOMATIC       145  155
138  0000H   1   PAGESIZE . . . .   BYTE EXTERNAL(57)         320
319  027CH  12   PAGESIZEROUT . .   PROCEDURE PUBLIC STACK=0010H
322  0288H  12   PAGEWIDTHROUT. .   PROCEDURE PUBLIC STACK=0010H
138  0000H   1   PASS . . . . . .   BYTE EXTERNAL(51)         146  326
  7              PCODEMACRO . . .   LITERALLY
  7              PCSEG. . . . . .   LITERALLY
  7              PDB. . . . . .   LITERALLY
  7              PDBIT. . . . . .   LITERALLY
  7              PDD. . . . . .   LITERALLY
  7              PDSEG. . . . . .   LITERALLY
  7              PDW. . . . . .   LITERALLY
  7              PEJECT . . . . .   LITERALLY
  7              PEND . . . . . .   LITERALLY
  7              PENDIF . . . . .   LITERALLY       139
  7              PENDM. . . . . .   LITERALLY
  7              PEQU . . . . . .   LITERALLY
  7              PESEG. . . . . .   LITERALLY
  7              PIF. . . . . .   LITERALLY       202
  7              PIFLIST. . . . .   LITERALLY
  7              PINCLUDE . . . .   LITERALLY
  7              PLIST. . . . . .   LITERALLY
  7              PMODRM . . . . .   LITERALLY
  7              PNOIFLIST. . . .   LITERALLY
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #

```
  7             PNOLIST. . . . . LITERALLY
  7             PNOSEGFIX. . . . LITERALLY
  7             PORG . . . . . LITERALLY
  7             PPAGESIZE. . . . LITERALLY
  7             PPAGEWIDTH . . . LITERALLY
  7             PRB. . . . . LITERALLY
  7             PRELB. . . . . LITERALLY
  7             PRELW. . . . . LITERALLY
112 0000H       PRINTCRLF. . . . PROCEDURE EXTERNAL(40) STACK=0000H
118 0000H       PRINTINIT. . . . PROCEDURE EXTERNAL(43) STACK=0000H
114 0000H       PRINTNEWPAGE . . . PROCEDURE EXTERNAL(41) STACK=0000H      355
138 0000H   1   PRINTON. . . . . BYTE EXTERNAL(52)       176  179  197  354  367  372  377
109 0000H       PRINTSINGLEBYTE. . PROCEDURE EXTERNAL(39) STACK=0000H
116 0000H       PRINTSOURCELINE. . PROCEDURE EXTERNAL(42) STACK=0000H      178  375
138 0000H   1   PRINTSWITCHOFF . . BYTE EXTERNAL(53)       364  366  376
120 0000H       PRINTTERMINATE . . PROCEDURE EXTERNAL(44) STACK=0000H
  2             PROC . . . . . LITERALLY       10   13   16   19   21   23   25   27   29   31   33
                                 35   37   39   41   43   45   48   50   53   74   77   86   89   92
                                 95   97   99  101  104  107  109  112  114  116  118  120  123  125
                                127  129  132  135  139  144  164  167  171  241  252  254  258  315
                                319  322  325  353  358  362  371  383  387
  7             PRS. . . . . LITERALLY
  7             PRW. . . . . LITERALLY
  7             PSEGFIX. . . . . LITERALLY
  4             PSEUDO . . . . . LITERALLY      187
  1 0000H       PSEUDOM. . . . . PROCEDURE STACK=0000H
  9             PSEUDOOPERR. . . . LITERALLY      160  347
172 0002H   1   PSEUDOTYPE . . . . BYTE EXTERNAL(60) AT      189  202
  7             PSIMFORM . . . . . LITERALLY
  7             PSSEG. . . . . LITERALLY
132 0000H   2   PT . . . . . WORD PARAMETER      133
129 0000H   2   PT . . . . . WORD PARAMETER      130
 65 0000H   2   PT . . . . . WORD PARAMETER       66
  7             PTITLE . . . . . LITERALLY
  6             RAH. . . . . LITERALLY
  6             RAL. . . . . LITERALLY
  6             RAX. . . . . LITERALLY
  6             RBH. . . . . LITERALLY
  6             RBL. . . . . LITERALLY
  6             RBP. . . . . LITERALLY
  6             RBX. . . . . LITERALLY
  6             RCH. . . . . LITERALLY
  6             RCL. . . . . LITERALLY
  6             RCS. . . . . LITERALLY
  6             RCX. . . . . LITERALLY
  6             RDH. . . . . LITERALLY
  6             RDI. . . . . LITERALLY
  6             RDL. . . . . LITERALLY
  6             RDS. . . . . LITERALLY
  6             RDX. . . . . LITERALLY
  2             REENT. . . . . LITERALLY
  4             REG. . . . . LITERALLY
  6             RES. . . . . LITERALLY
 35 0000H       REWINDSOURCE . . . PROCEDURE EXTERNAL(11) STACK=0000H
  8             RIGHTBRACKET . . . LITERALLY
  6             RSI. . . . . LITERALLY
  6             RSP. . . . . LITERALLY
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950
SER. #_____

```
  6                RSS. . . . . .   LITERALLY
 83  0000H    2   S. . . . . .     WORD PARAMETER       84
 56  0000H    2   S. . . . . .     WORD PARAMETER       57
 97  0000H        SCAN . . . . . .  PROCEDURE EXTERNAL(34) STACK=0000H      186  194  278  279  287  290
                                     297  343
 95  0000H        SCANINIT . . . . .  PROCEDURE EXTERNAL(33) STACK=0000H
  3                SEGMBIT. . . . . .  LITERALLY
  3                SEGMCOUNT. . . . .  LITERALLY
 1(  0004H    2   SEGMENT. . . . . .  WORD MEMBER(OPER)
145  0004H    2   SEGMENT. . . . . .  WORD MEMBER(OPER)
138  0004H    2   SEGMENT. . . . . .  WORD MEMBER(OPERANDS)
  3                SEGTYPEBIT . . . .  LITERALLY
  3                SEGTYPECOUNT . . .  LITERALLY
166  0003H    1   SFLAG. . . . . .  BYTE MEMBER(OPER)
14[  0003H    1   SFLAG. . . . . .  BYTE MEMBER(OPER)
138  0003H    1   SFLAG. . . . . .  BYTE MEMBER(OPERANDS)
138  0000H    1   SIMFORM. . . . . .  BYTE EXTERNAL(58)      359
353  0307H   13   SIMFORMROUT. . . .  PROCEDURE PUBLIC STACK=000AH
                  SIZE . . . . . .  BUILTIN       274
144  001AH   54   SIZEWIDTH. . . . .  PROCEDURE STACK=000CH       320  323
 99  0000H        SKIPRESTOFLINE . .  PROCEDURE EXTERNAL(35) STACK=0000H      142  148  161  169  208  214
                                     234  249  269  306  335  338  348
104  0000H        SKIPUNTIL. . . . .  PROCEDURE BYTE EXTERNAL(37) STACK=0000H
171  00A7H  114   SKIPUNTILENDIF . .  PROCEDURE STACK=000AH       235
138  0000H    1   SOURCEPTR. . . . .  BYTE EXTERNAL(59)       368
  2                SPACE. . . . . .  LITERALLY       274
  4                SPEC . . . . . .  LITERALLY
101  0000H        SPECIALTOKEN . . .  PROCEDURE BYTE EXTERNAL(36) STACK=0000H      288
  4                STRING . . . . . .  LITERALLY       328  341
  2                STRUC. . . . . .  LITERALLY       138  145  166
166  0002H    1   STYPE. . . . . .  BYTE MEMBER(OPER)       222
145  0002H    1   STYPE. . . . . .  BYTE MEMBER(OPER)       153
138  0002H    1   STYPE. . . . . .  BYTE MEMBER(OPERANDS)
  4                SYMBOL . . . . . .  LITERALLY
  3                SYMBOLHEAD . . . .  LITERALLY
  3                SYMBOLSTRUC. . . .  LITERALLY       138  145  166
254  0230H   16   SYNTAXERR. . . . .  PROCEDURE STACK=0006H       283  293
 59  0000H    2   T. . . . . .      WORD PARAMETER       60
  2                TAB. . . . . .    LITERALLY
174  0002H    1   TEMP . . . . . .  BYTE      180  183  196
139  0000H   23   TESTEMPTYLINE. . .  PROCEDURE STACK=0006H       156  195  231  245  311  316  344  356
                                     360  363  374  380  384  388
 62  0000H    2   TEXTADR. . . . . .  WORD PARAMETER       63
  2                THENDO . . . . . .  LITERALLY       146  151  153  176  187  189  192  202  211  217  222
                                     228  242  266  275  281  288  291  304  328  341  364  372
138  0000H   30   TITLE. . . . . .  BYTE ARRAY(30) EXTERNAL(56)       330  331  332  333
325  0294H   97   TITLEROUT. . . . .  PROCEDURE PUBLIC STACK=000AH
104  0000H    1   TOK. . . . . .    BYTE PARAMETER       105
101  0000H    1   TOK. . . . . .    BYTE PARAMETER       102
138  0000H    4   TOKEN. . . . . .  STRUCTURE EXTERNAL(60)       172  187  328  341
  2                TRUE . . . . . .  LITERALLY       180  197  256  262  312  317  359  367  376  385
138  0000H    1   TYPE . . . . . .  BYTE MEMBER(TOKEN)       187  328  341
  3                TYPEBIT. . . . . .  LITERALLY
 50  0000H        TYPECALC . . . . .  PROCEDURE BYTE EXTERNAL(18) STACK=0000H
  3                TYPECOUNT. . . . .  LITERALLY
  4                UDEFSYMB . . . . .  LITERALLY
  9                UDEFSYMBOL . . . .  LITERALLY
```

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P.O. Box 579
PACIFIC GROVE, CA 93950
USER # _____

```
 30  0000H        UPPER. . . . . .   PROCEDURE BYTE EXTERNAL(28) STACK=0000H
120  0000H    1   USEFACT. . . . .   BYTE PARAMETER       121
 50  0000H    2   VAL. . . . . . .   WORD PARAMETER        51
138  0002H    2   VALUE. . . . . .   WORD MEMBER(TOKEN)        172
  4                VARIABLE . . . .   LITERALLY
  5                WRD. . . . . . .   LITERALLY
 53  0000H        WRDTEST. . . . .   PROCEDURE BYTE EXTERNAL(19) STACK=0000H
```

MODULE INFORMATION:

```
CODE AREA SIZE     = 036DH    877D
CONSTANT AREA SIZE = 0003H      3D
VARIABLE AREA SIZE = 0011H     17D
MAXIMUM STACK SIZE = 0010H     16D
862 LINES READ
0 PROGRAM ERROR(S)
```

END OF PL/M-86 COMPILATION

COPYRIGHT © 1981, 1982
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950

SER. #_____